NAME

XGetDeviceProperty, XChangeDeviceProperty, XDeleteDeviceProperty - Get, change or delete a device's property.

SYNOPSIS

#include <X11/extensions/XInput.h>

int XGetDeviceProperty(Display *display,
                XDevice *device,
                Atom property,
                long offset,
                long length,
                Bool delete,
                Atom req_type,
                Atom *actual_type_return,
                int *actual_format_return,
                unsigned long *nitems_return,
                unsigned long *bytes_after_return,
                unsigned char **prop_return)

void XChangeDeviceProperty(Display *display,
                XDevice *device,
                Atom property,
                Atom type,
                int format,
                int mode,
                const char *data,
                int nelements)

void XDeleteDeviceProperty(Display *display,
                XDevice *device,
                Atom property)

actual_type_return
     Returns an atom identifier that defines the actual type
     of the property.

actual_format_return
     Returns the actual format of the property.

bytes_after_return
> Returns the number of bytes remaining to be read in the
> property if a partial read was performed.

data
> Specifies the property data.

delete
> Specifies a Boolean value that determines whether the
> property is deleted.

display
> Specifies the connection to the X server.

device
> The device to grab.

format
> Specifies whether the data should be viewed as a list of
> 8-bit, 16-bit, or 32-bit quantities. Possible values are
> 8, 16, and 32. This information allows the X server to
> correctly perform byte-swap operations as necessary. If
> the format is 16-bit or 32-bit, you must explicitly cast
> the data pointer to an (unsigned char*) in the call to
> XChangeDeviceProperty.

length
> Specifies the length in 32-bit multiplies of the data to
> be retrieved.

mode
> Specifies the mode of operation. You can pass
> PropModeReplace, PropModePrepend, or PropModeAppend.

nelements
> Specifies the number of elements in data.

nitems_return
> Returns the actual number of 8-bit, 16-bit, or 32-bit
> items stored in the prop_return array.

num_values
> Specifies the number of elements in the values list.

offset
> Specifies the offset in the specified property (in
> 32-bit quantities) where the data is to be retrieved.

property
> Specifies the property to modify or query.

prop_return
> Returns the data in the specified format. If the
> returned format is 8, the returned data is represented
> as a char array. If the returned format is 16, the
> returned data is represented as an array of short int
> type and should be cast to that type to obtain the
> elements. If the returned format is 32, the property
> data will be stored as an array of longs (which in a
> 64-bit application will be 64-bit values that are padded
> in the upper 4 bytes).

req_type
> Specifies the atom identifier associated with the
> property type or AnyPropertyType.

DESCRIPTION

The XGetDeviceProperty function returns the actual type of the property; the actual format of the property; the number of 8-bit, 16-bit, or 32-bit items transferred; the number of bytes remaining to be read in the property; and a pointer to the data actually returned. For a detailed description of this function, see the man page to XGetWindowProperty.

The XChangeDeviceProperty function alters the property for the specified device and causes the server to generate a XPropertyNotifyEvent event on that device. For a detailed description of this function, see the man page to XChangeProperty.

The XDeleteDeviceProperty function deletes the specified device property. Note that a client cannot delete a property created by a driver or the server. Attempting to do so will result in a BadAtom error.

XGetDeviceProperty can generate a BadAtom, BadDevice error.

XChangeDeviceProperty can generate a BadDevice, a BadMatch, a BadAtom, and a BadValue error.

XDeleteDeviceProperty can generate a BadDevice, and a BadAtom error.

DIAGNOSIS

BadAtom
> A value does not describe a valid named identifier or the client attempted to remove a driver-allocated property.

BadDevice
> An invalid device was specified. The device does not exist.

SEE ALSO

XListDeviceProperties(3)